

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

**A REPORT ON
THE ONLINE FRUIT WEIGHING, IMAGING AND SORTING MECHANISM**



Prepared by:

SHUBH VACHHER,	2014A8PS471G
KASTURI ROOPA AKHILA,	2014AAPS250H

THIS REPORT HAS BEEN PREPARED AS DOCUMENTATION FOR THE PROJECT ON
ONLINE FRUIT SORTING MECHANISM AS A PART OF THE PRACTICE SCHOOL I
PROGRAMME.

PS STATION: CENTRAL ELECTRONICS ENGINEERING RESEARCH
INSTITUTE (CEERI), CHENNAI

PS INSTRUCTOR: Mr G.MEENAKSHI SUNDARAM

PROJECT LEADER: Mr. MOHAMMED IQBAL (SENIOR PRINCIPAL SCIENTIST AND
DEPUTY DIRECTOR, CEERI)

DATE OF SUBMISSION: 15-07-2016

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

**A REPORT ON
THE ONLINE FRUIT WEIGHING, IMAGING AND SORTING MECHANISM**

PREPARED BY

SHUBH VACHHER, 2014A8PS471G

KASTURI ROOPA AKHILA 2014AAPS250H

THIS REPORT HAS BEEN PREPARED AS DOCUMENTATION FOR
THE PROJECT ON ONLINE FRUIT SORTING MECHANISM AS A PART OF THE
PRACTICE SCHOOL I PROGRAMME.

PS STATION: CENTRAL ELECTRONICS ENGINEERING RESEARCH
INSTITUTE (CEERI), CHENNAI

PS INSTRUCTOR: Mr G.MEENAKSHI SUNDARAM

PROJECT LEADER: Mr. MOHAMMED IQBAL (SENIOR PRINCIPAL SCIENTIST AND
DEPUTY DIRECTOR, CEERI)

DATE OF SUBMISSION: 15-07-2016

**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
PILANI (RAJASTHAN)
PRACTICE SCHOOL DIVISION**

PS STATION: CENTRAL ELECTRONICS ENGINEERING RESEARCH INSTITUTE
PS CENTRE: CHENNAI
DURATION: **FROM:** 23/5/2016 **TO:** 15/7/2016

DATE OF SUBMISSION: 15/7/2016

TITLE OF THE PROJECT: ONLINE FRUIT WEIGHING, IMAGING AND SORTING
MECHANISM

SHUBH VACHHER **2014AAPS471G**
B.E (ELECTRONICS AND INSTRUMENTATION ENGINEERING)
KASTURI ROOPA AKHILA **2014AAPS250H**
B.E (ELECTRONICS AND COMMUNICATION ENGINEERING)

NAME OF EXPERT: MOHAMMED IQBAL
DESIGNATION: SENIOR PRICIPAL SCIENTIST, DEPUTY DIRECTOR, CEERI, CHENNAI

NAME OF THE PS INSTRUCTOR: Mr. G. MEENAKSHI SUNDARAM

KEY WORDS: ARDUINO, IMAGE PROCESSING, OPENCV, PYTHON, FESTO VALVES,
OPTICAL PROXIMITY SWITCH, WEIGHING SCALE, USB CAMERA, LIGHTING AND
IMAGING STATION, SERIAL COMMUNICATION

ABSTRACT:

The project is concentrated mainly on grading the fruits (Sweet limes) into three different grades: A, B, C based on their sizes (radii). The fruits move along a conveyor belt where they get imaged by a USB camera and the image is processed by a laptop and the laptop returns the grade of the fruit to the micro controller, an Arduino. The Arduino thus triggers the tilting platform, such that the platform tilts into a particular direction for each particular grade. The second part of the project deals with the weighing of the fruits that pass down the conveyor belt.

SIGNATURES OF THE STUDENTS

SIGNATURE OF THE PS INSTRUCTOR

ACKNOWLEDGEMENTS:

I express my sincere gratitude to our mentor Mr. Mohammed Iqbal, Senior Principal Scientist, Deputy Director, CEERI, Chennai for providing us with all the necessary guidance and freedom to work on the project allotted.

I would like to thank our PS Instructor Mr.G.Meenakshi Sundaram for being friendly and encouraging all along the course.

I would also like to express my gratitude to Mrs.Thilaka Mohandoss, Student Co-ordinator, CEERI, Chennai for extending her co-operation all along the way.

I would also like to appreciate the help extended by Mrs. Uma, Principal Technical Officer and Mrs.Meenakshi in the hardware part of our project.

I would also thank the technical staff Mr.Chandra Moorthy and Mr.Ravi for helping us in fixing all our problems.

SIGNATURES OF THE STUDENTS

SHUBH VACHHER

2014A8PS471G

KASTURI ROOPA AKHILA

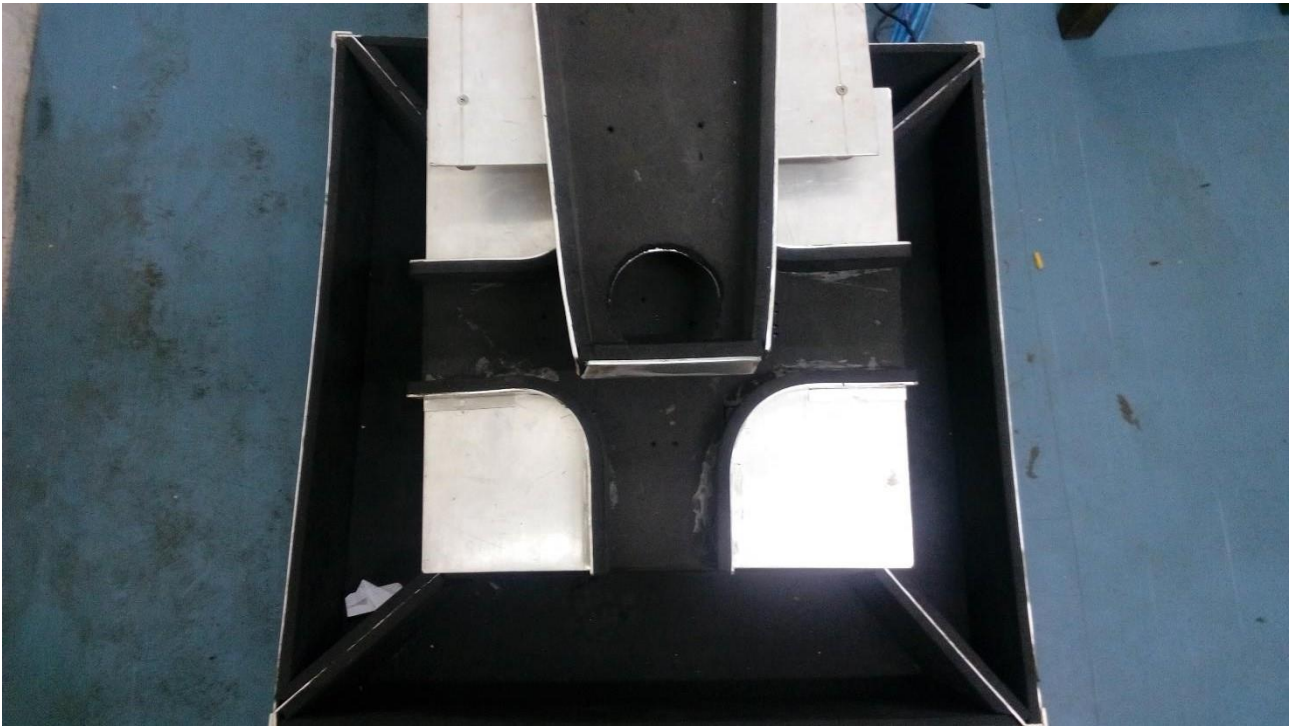
2014AAPS250H

CONTENTS

• ONLINE FRUIT IMAGING AND SORTING	7
1. The Coding	8
2. The Image Processing	10
3. The Electronics	10
▪ Arduino	10
▪ USB to UART	11
▪ IC-ULN-2803A	12
▪ Photo-Electric Sensor	13
▪ Infrared Sensor	14
▪ Optical Proximity Switch	15
4. Conclusion	19
• AUTOMATED STATIONARY WEIGHING MECHANISM	22
1. About the Code and Connections	22
2. Conclusion	24
• Bibliography	25

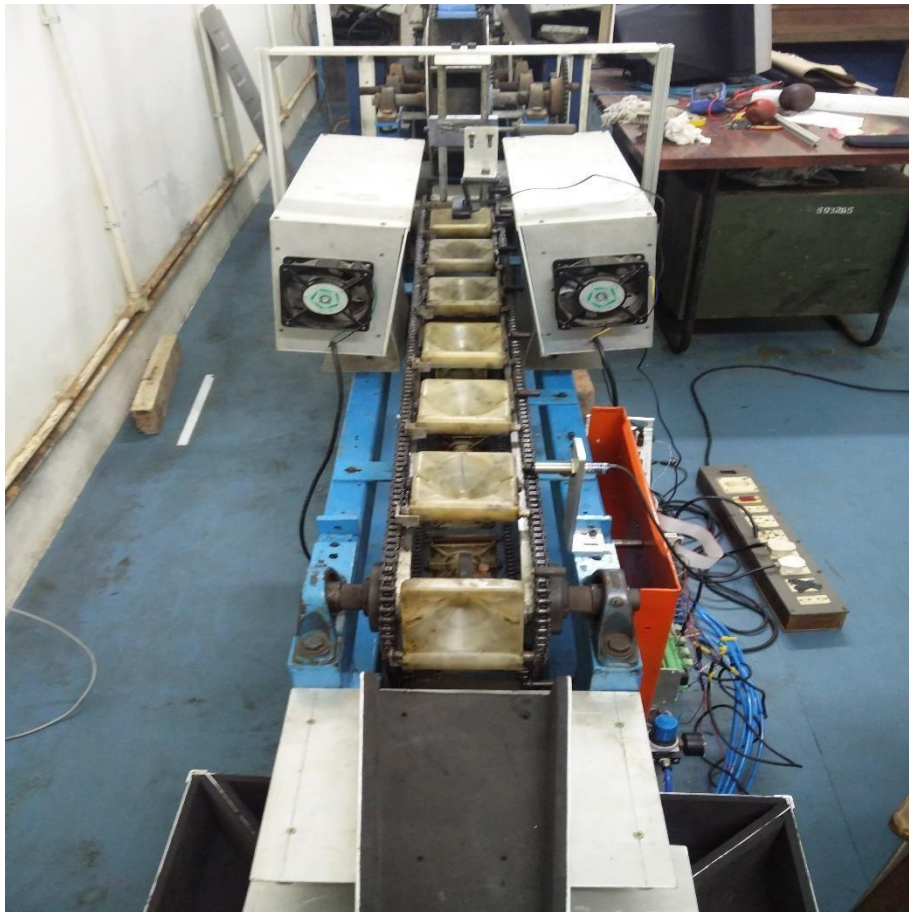
ONLINE FRUIT IMAGING AND SORTING

FIGURE 1: An image of the tilting platform -



Thus based upon the grade of the fruit, the fruit gets sorted.

FIGURE 2: The whole system can be shown in a single picture as –



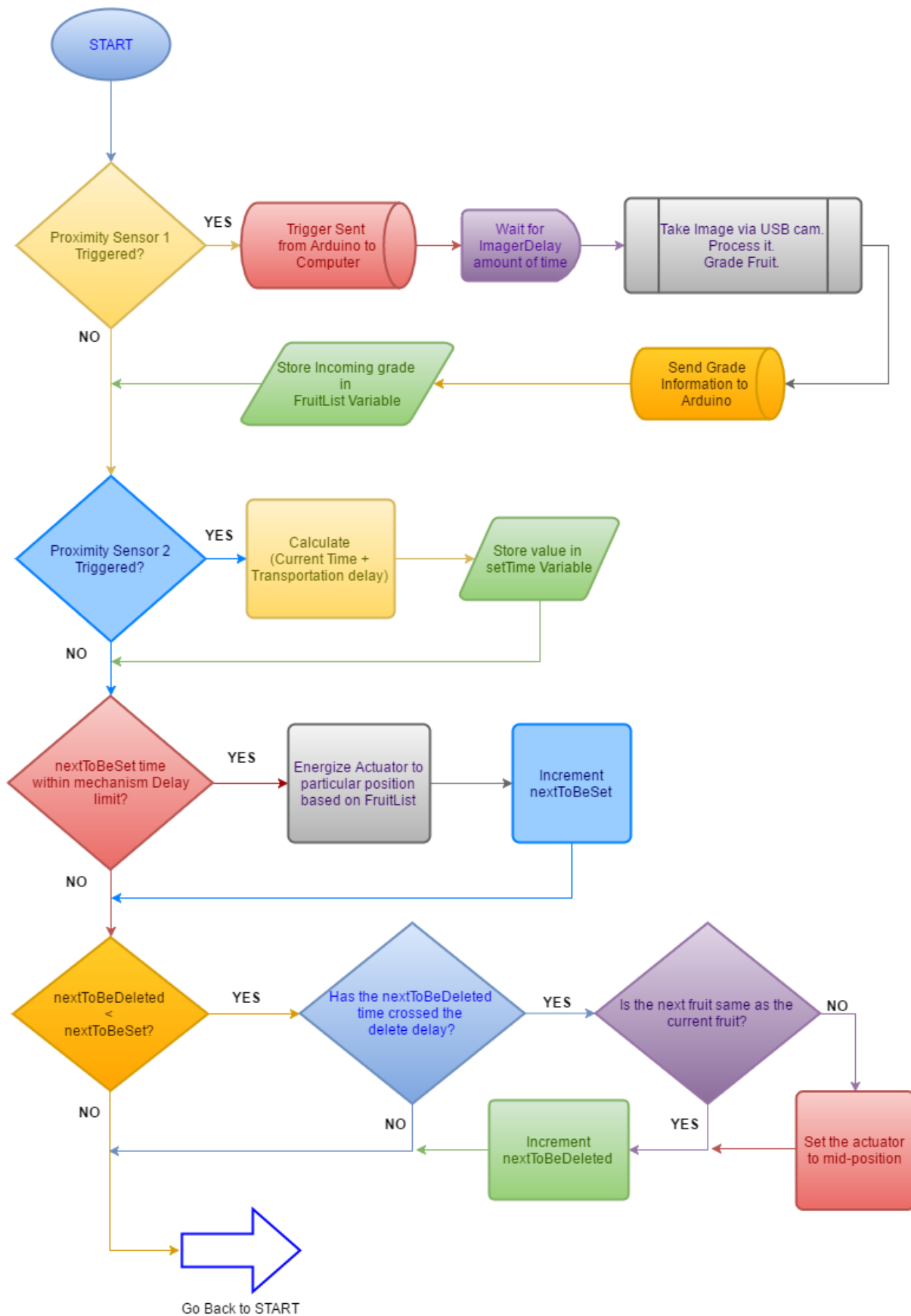
THE CODING:

Our project runs entirely based on two codes. One code is written in the Arduino language running on the Arduino software (which needs to be uploaded on to the Arduino) and the other one written in python running on the laptop.

The code running on the Arduino basically takes inputs from the optical proximity switch placed right before the lighting and the imaging stations and sends them to the laptop so that the laptop knows exactly when to trigger the USB camera for the camera to take images. It also receives the fruit grade data from the laptop (after the image processing is done) and stores it in the fruitList variable. This code also checks the inputs from the photo-electric sensor that has been placed at the end of the conveyor belt so that it knows exactly when to service a particular fruit by tilting the platform.

The code running on the laptop ,written in python takes the inputs from the Arduino and triggers the USB camera to capture the image of the fruit and then processes it and returns the grade of the respective fruit to the Arduino.

FIGURE 3: The entire functioning of these two codes can be explained with the help of the below flow chart-



THE IMAGE PROCESSING:

The image processing is done using Open Cv written in python language. This is done by the python code running on the laptop as mentioned in the above section under coding. The basic aim of the image processing done in this project is just to extract the radius of the fruit from the image. The code takes the image and converts it into a black and white image, then does an adaptive thresholding on the black and white image. It then prepares a mask, to delete the extras in the image-like the basket, railings of the conveyor, insides of the fruit etc., and then uses this mask on the thresholded image. Now the contours in this new masked image are obtained that give us the just the edges of the fruit. The radius of this contour (edge of the fruit) is thus obtained from the area of the contour which is found an in built function in Open Cv – contourArea. In fact most of the things explained above are obtained using in built methods and functions in Open Cv.

THE ELECTRONICS:

Coding and Electronics go hand in hand in our project. As the detailing has to be started somewhere, we just started out with coding. But the electronics also plays an equally important role in the functioning of the system. The electronics part is divided in to modules so that it would be simpler for the reader to understand the basic operation.

The first and the foremost module would be the Arduino.

ARDUINO:

We have used an Arduino MEGA ADK board for all the purposes in this project.

FIGURE 4: Image of an Arduino



The Arduino acts as a medium that takes the inputs from all the sensors that are used in the system and provides the appropriate responses to the corresponding inputs.

The Arduino is powered by a 12V power source and is connected to the laptop via a USB to UART converter. Thus the laptop is just connected only for Serial communication and not for powering up the Arduino. The Arduino is connected to both the optical proximity switches and also to the level shifter IC also to the level shifter IC ULN 2803A by means of the several Digital and Analog pins available on it. All these modules will be explained in a greater detail in the following sections.

The next module would be the USB to UART converter.

USB TO UART CONVERTER:

This module is used in our system in order to attain stable serial communication between the PC and the Arduino board without allowing the PC to power up the Arduino

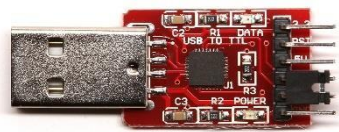


FIGURE 4:

It looks similar to the one shown in the picture.

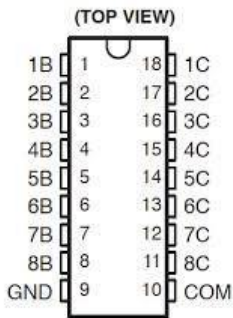
The module has six pins and they are 3V3, GND, VBUS, TXD, RXD, DTR (top to bottom).

The DTR pin is used to RESET the Arduino during upload time and hence is connected to the RESET pin of the Arduino with a capacitor in between (for proper RESET operation).

The RXD (Receive Output) pin is connected to the TX pin on the Arduino board. The TXD (Transmit Output) pin is connected to the RX pin on the Arduino board.

The GND pin is connected to the GND pin of the Arduino. The VBUS and the 3V3 pin are left unused.

The next module would be that of the level shifter IC.



IC ULN 2803A:

This IC steps up(shifts the level- as the name suggests) the outputs of the Arduino(0/5V) and gives outputs that are either zero or 24V. These outputs are fed to the relays which are in turn connected to the festo valves.

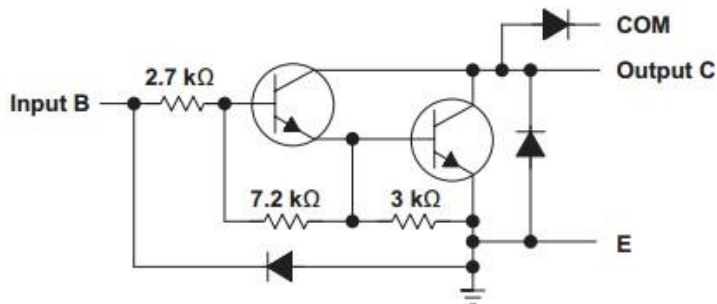
The 22-29 pins of the Arduino are connected to the 1-8 pins of the IC.

FIGURE 5: The IC is similar to the one shown in the picture.

This picture shows the exact pinout of the IC. The pins 1B-8B are connected to the 22-29 pins of the Arduino. The pins 1C-8C go to the relays via a connector.

The IC has 8 Darlington pairs (with NPN transistors) and the pins 1B-8B are the 8 base inputs each corresponding to a particular Darlington pair. The pins 1C-8C are the collector outputs corresponding to a particular Darlington pair.

FIGURE 6: The picture of a single Darlington pair is shown below.



The next module would be that of the sensors.

Deciding on what sensor to use for a particular job had been tough. So we ended up choosing the wrong ones. All the sensors that have been used in our system including the ones that have been replaced are explained in a great detail in the current section.

In order to trigger the USB camera to take the images of the fruits, a Photo Electric Sensor has been used in the beginning.

PHOTO ELECTRIC SENSOR:



FIGURE 7: The photo electric sensor that we have used in our system is a PZ2 51T sensor.

It has two components - one transmitter and a receiver.

The RED LED on the receiver is for OUTPUT and POWER and the GREEN LED on the receiver indicates stable operation.

The RED LED turns off when some obstacle comes in between the transmitter and the receiver. The BROWN (or RED) wires from both the transmitter and the receiver go to the 12V of the power supply and the BLUE wires from both the transmitter and the receiver go to the GND of the power supply. The BLACK wire of the receiver gives the output. The output is HIGH active, i.e., it goes HIGH whenever there is an obstacle between the transmitter and the receiver. This wire is connected to an Analog pin on the Arduino board (we have used the A1 pin in our system, verify with the code).

The pink wire of the transmitter is unused.

FIGURE 8: The working of this sensor is clearly depicted in the picture attached below.



Drawbacks of using this sensor: The PE sensor detects just the knobs present on the edge of each basket on the conveyor. This makes the image processing part a bit too complicated as the camera cannot differentiate between a basket with a fruit and the one without it. Hence even the empty baskets are imaged and extra care is required to be taken while we code for the image processing.

In order to make life easy, we chose to replace the PE sensor with an optical proximity switch which can be used to detect the presence of the fruit in the basket so that the USB camera can be triggered only when there is a fruit in the basket.

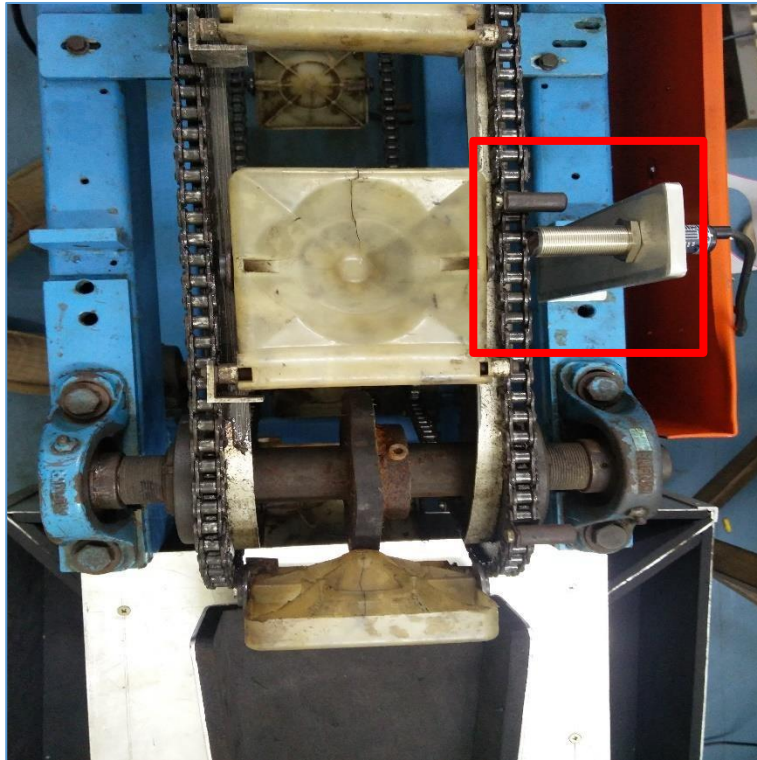


FIGURE 9: This optical proximity switch in our system

The other sensor that we used is an IR sensor. This sensor is used to detect the presence of the fruit in the basket so that the Arduino would know exactly when to service a fruit.

INFRARED SENSOR:

FIGURE 10: The IR sensor used in our system is similar to the one that is shown in the picture.

The IR sensor has three pins – 5V, GND, OUT pins.

The 5V pin is connected to the 5V pin of the Arduino board. This helps in powering up the sensor.

The GND pin is connected to the GND of the Arduino board. The OUT pin is connected to any of the Analog pins on the Arduino. We have used the A4 pin for that purpose and should be verified from the code.

This sensor is used in our module as it is inexpensive, easily available and simple to connect.

Drawbacks of using this sensor: When an IR sensor is used, one faces the disadvantage of external light as a stimulus. The IR sensor gives absurd results in the presence of light and hence it is difficult to get proper results out of it when one wishes to use it in broad daylight. One can always use a different sensor in place of the IR sensor and the connections exactly similar to that of the



IR sensor can be made. The OUT pin of the sensor must be connected to any one of the Analog pins just as in the case of the IR sensor.

We hence chose to replace the IR sensor with a simple optical proximity switch. The optical proximity switch is thus used to detect the presence of the fruit in the basket at the end of the conveyor so that the Arduino knows exactly when to tilt the platform. We thus ended up having two optical proximity switches in our system. One replacing the Photo electric sensor before the imaging station and the other replacing the IR sensor at the end of the conveyor.

OPTICAL PROXIMITY SWITCH:



FIGURE 11: The proximity switch that is used in our system is similar to the one that is shown in the figure.

This sensor has four wires and they are RED, BLUE, WHITE and BLACK.

The RED and BLUE wires of the sensor are connected to 5V and GND pins of the Arduino respectively. The BLACK wire gives an output of 5V when an object is sensed by the sensor and is connected to one of the Analog pins. The WHITE wire is left unused.

This sensor is chosen as it is highly accurate and easily available. As all the important modules of the system have been explained in an explicit manner, we move on to showing the detailed connections of the hardware

FIGURE 12 : DETAILED CONNECTIONS OF THE HARDWARE

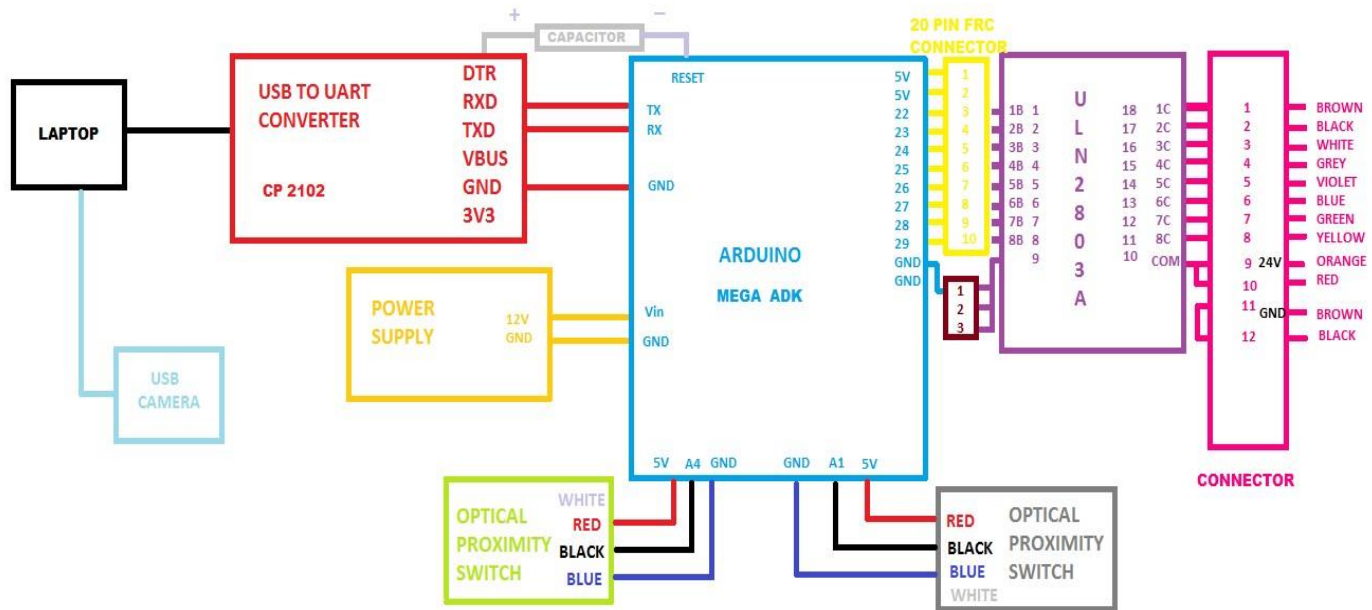


FIGURE 13 : CONNECTIONS OF THE RELAYS TO THE FESTO VALVES

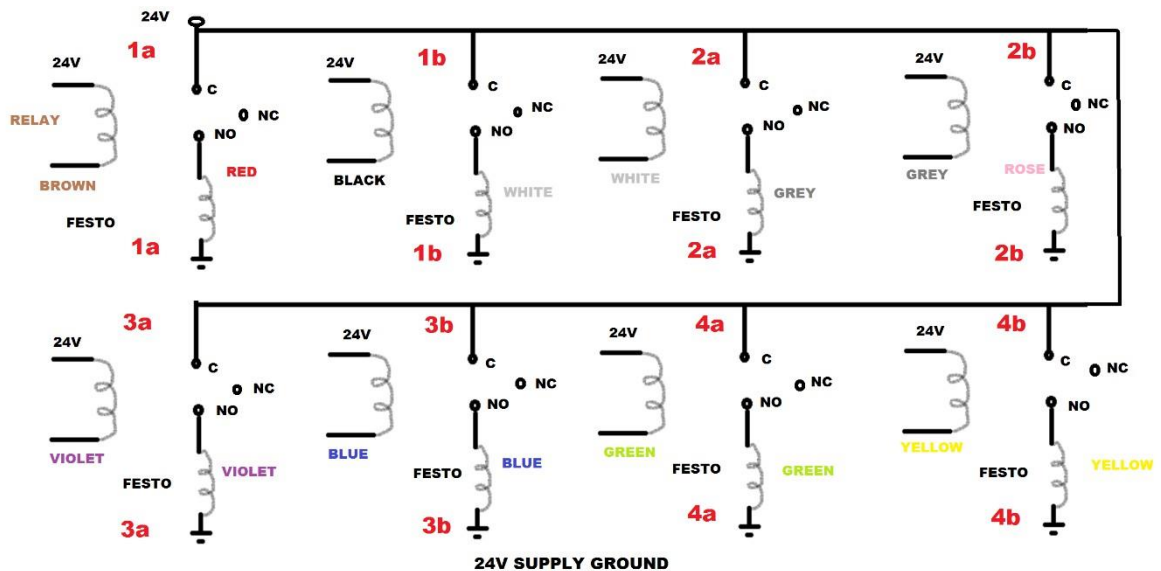
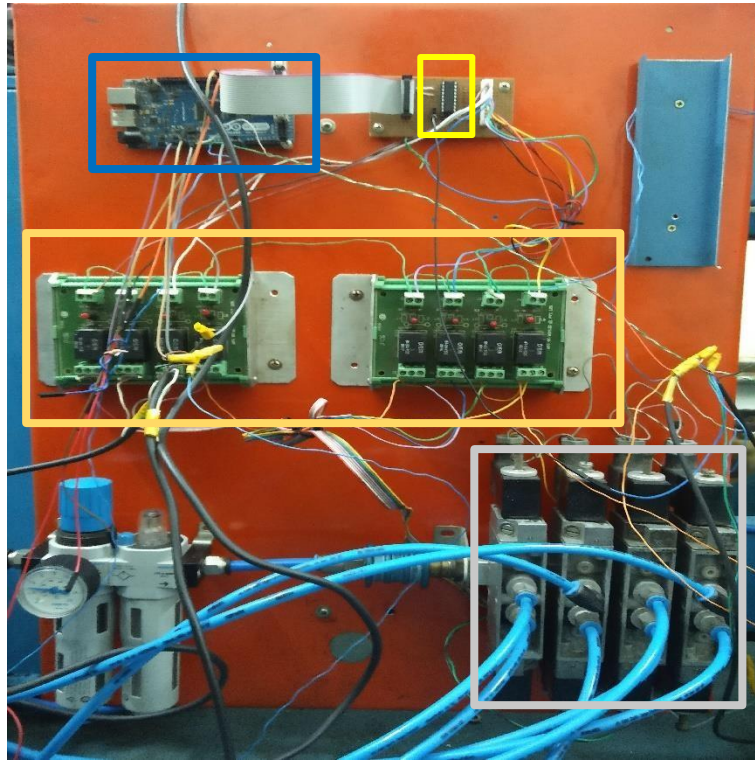


FIGURE 14:Both of the above connections in our real life project can be seen from the picture attached below –



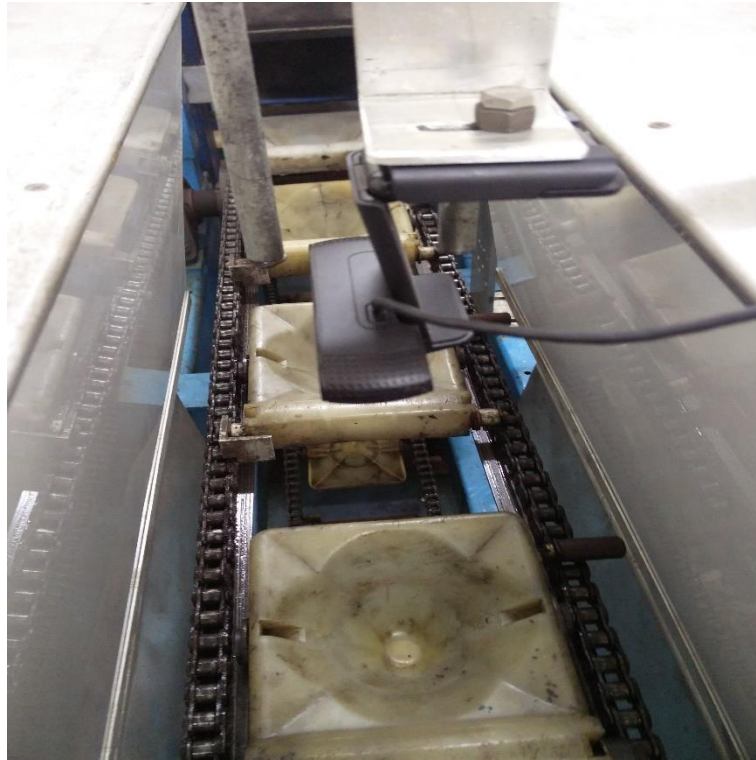
The one highlighted in the BLUE coloured box is the Arduino, the one in the YELLOW coloured box is the IC ULN 2803A, the ones in the CREAM coloured box are the relays and the ones in the GREY coloured box are the festo valves.

For a clear understanding of the system, we hereby attach the pictures of the imaging and lighting stations.

FIGURE 15:The lighting station is as shown in the below picture-



FIGURE 16: The USB camera module is as shown in the below picture-



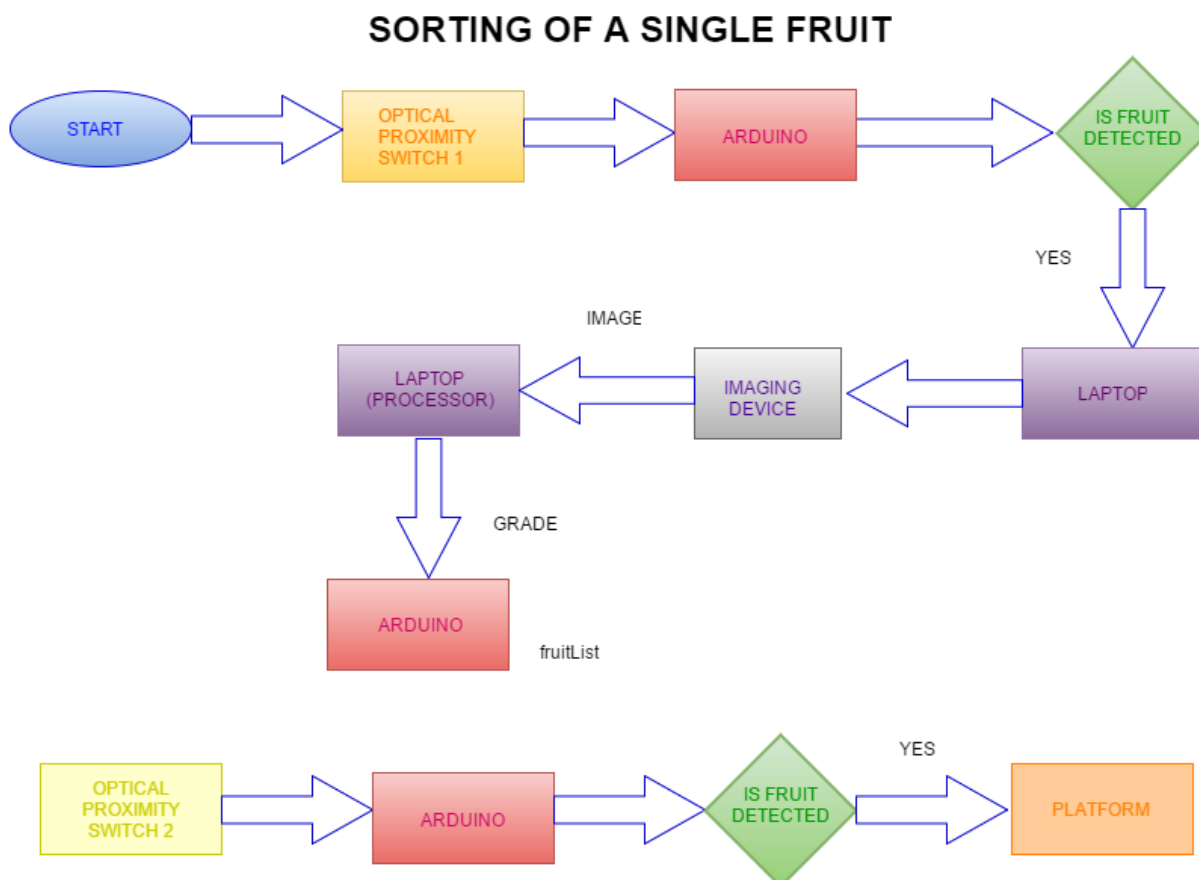
CONCLUSION:

A final flowchart has been attached that gives us the complete operation of the system.

This flowchart is made with the following assumptions:

- The python code is running on the laptop and the Arduino code has been uploaded on to the Arduino.
- Python code is running on the laptop at current time.
- The laptop is connected to both Arduino and the imaging device.

FIGURE 17: FLOWCHART DEPICTING THE COMPLETE OPERATION OF THE SYSTEM



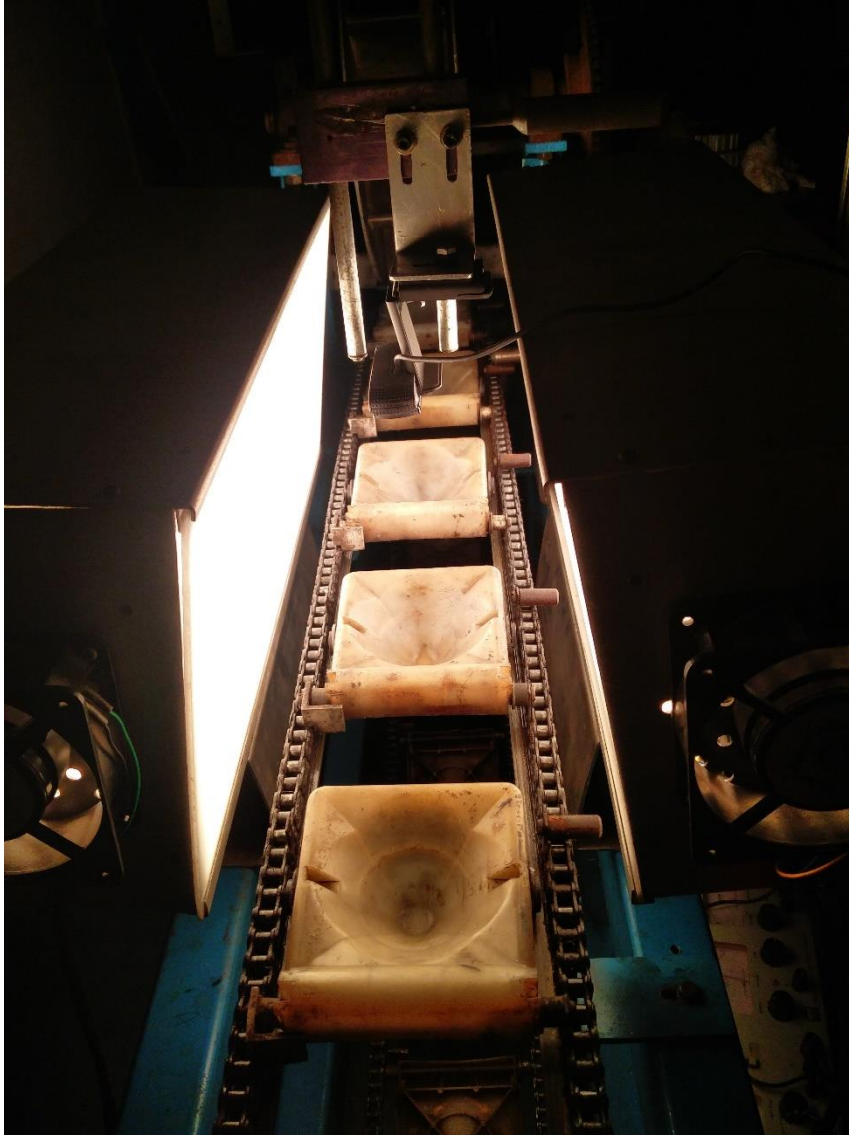
The flow of the system is as follows:

1. The optical proximity switch (the one used instead of the PE sensor) input is read by the Arduino.
2. When a Fruit Holder is detected to pass the sensor, a signal is sent to the computer via the Arduino.
3. The computer then activates the USB Camera and takes as well as processes the image taken.
4. A grade of the imaged fruit is sent back to the Arduino and is stored in the fruitList variable.
5. Now the Arduino waits for the optical proximity switch (the one used instead of the IR sensor) to detect if a fruit has crossed it.
6. Using the fruitList variable, the Arduino knows which fruit has crossed the optical proximity switch and accordingly sets the grading platform to sort the fruit into the correct grade basket.

FIGURE 18: So, our total system in action is shown in the images attached-



FIGURE 19: The lighting and imaging stations



AUTOMATED STATIONARY WEIGHING MECHANISM

The second part of the project deals with the weighing of the fruits that pass down the conveyor belt. The fruits pass down the conveyor belt and fall onto one of the pockets of the stationary weighing platform. The fruits are then indexed onto the next pocket which has an electronic weighing scale. The values outputted from the weighing scale are given to an Arduino that has a program running on it. The program takes the outputs of the weighing scale and takes only values into consideration that are in a particular range.

FIGURE 20: The entire flow of this process is explained by the following flow chart-

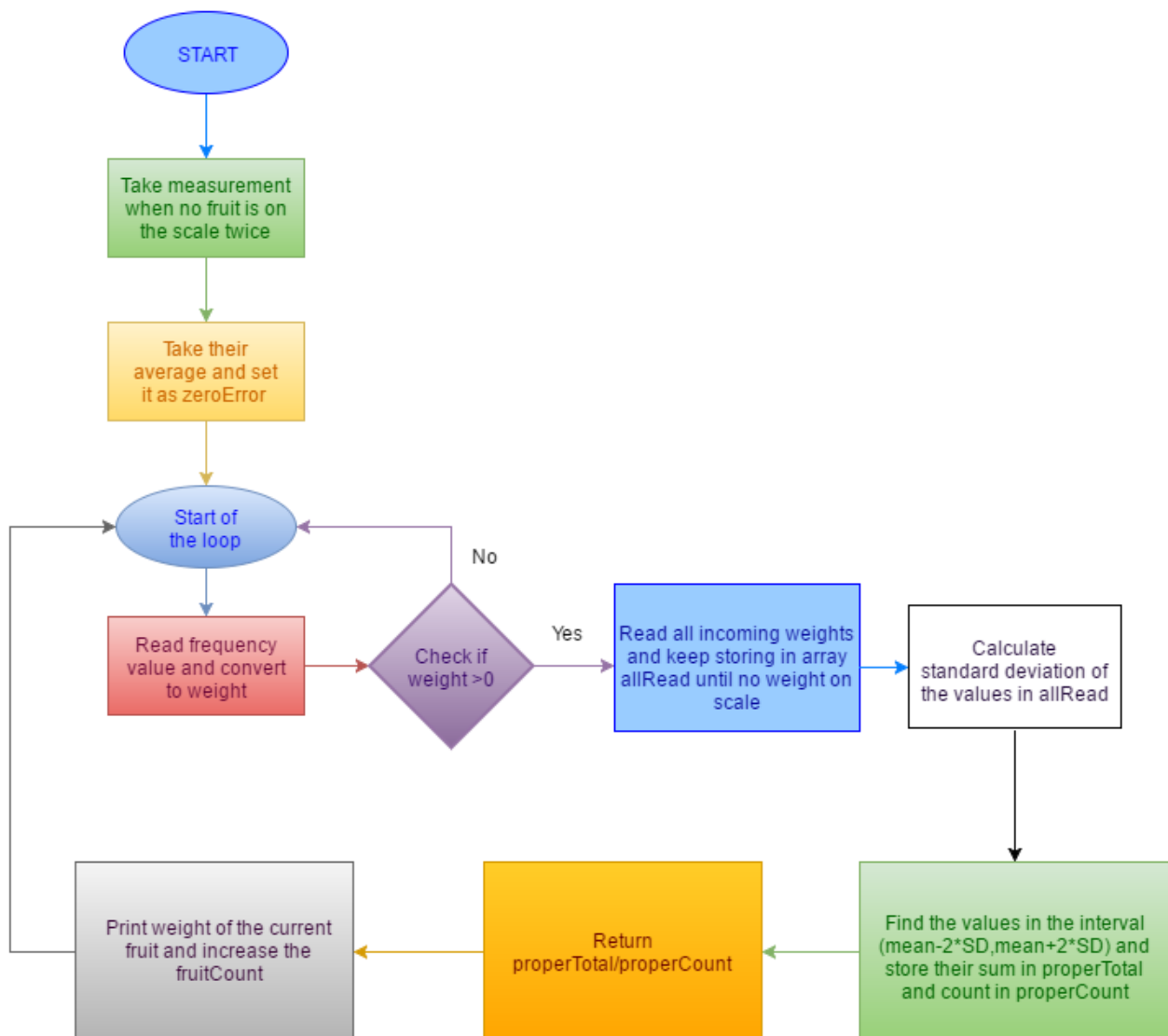


FIGURE 21: The picture of the weighing platform is attached here-

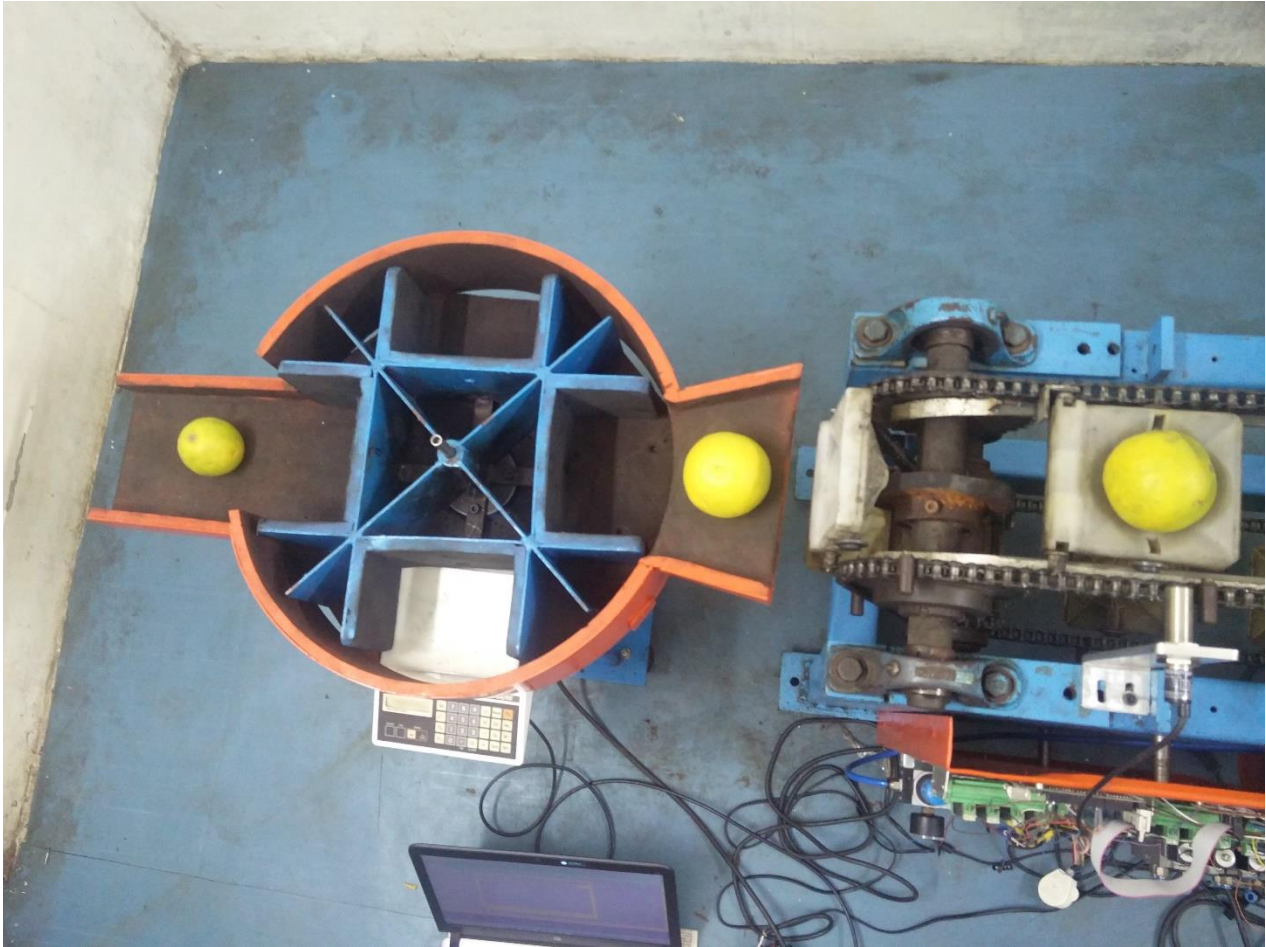


FIGURE 22: The image of the entire system is attached in here-



CONCLUSION

The fruits thus fall from the conveyer onto the weighing scale and thus get weighed successfully.

Our project on the Online Fruit Imaging, Sorting and Grading has thus been explained.

Bibliography

Most of our sources have been online. We have acknowledged them below

- Stack Overflow
- PyImageSearch
- Wikipedia
- OpenCV Manuals
- Python Manuals
 - Let us Python
- Arduino Library
- Keyence Datasheets

SIGNATURES OF THE STUDENTS:

- | | |
|---|--------------|
| <input type="checkbox"/> SHUBH VACHHER | 2014A8PS471G |
| <input type="checkbox"/> KASTURI ROOPA AKHILA | 2014AAPS250H |